Same Form, Different Force: Outcome-Stable Interpretation in LLMs

Anonymous ACL submission

Abstract

Utterances with the same surface form can preserve a stable outcome space while differing in role assignment, responsibility, and interactional force. We study this problem through outcome-stable discourse-resolved utterances, where grammar constrains the event at issue, but discourse determines who acts, who is responsible, and what social act the utterance performs. We formalize interpretation using an Outcome–Role–Attitude decomposition, $I(c, u) = (O(u), R(c, u), A(c, u))$, and examine South Asian language cases where directive, causative, and elliptical forms shift between instruction, request, delegation, competence attack, and humiliation depending on context. We evaluate LLMs on context-controlled minimal pairs and find that models are comparatively stable on outcome-space but less reliable on discourse-resolved role and attitude. These results show that the same form can carry different force, and that LLM interpretation requires separating outcome stability from role and attitude resolution.

1 Introduction

Large language models are often described as learning linguistic constructions: reusable form–meaning pairings that support generalization across utterances. This is a natural hypothesis. On the other hand, constructional structure is real, and construction-based accounts have long treated linguistic knowledge as organized around pairings of form and meaning (Fillmore et al., 1988; Goldberg, 1995, 2006; Croft, 2001). Recent work has also begun to ask whether neural language models encode, use, or generalize constructional information (Weissweiler et al., 2023; Tayyar Madabushi et al., 2023; Scivetti et al., 2025). But this constructional learning is not enough to explain a class of cases where the form remains fixed while the force of the utterance changes. If the same utterance can preserve its surface structure and broad outcome while shifting role assignment, responsibility, or stance across discourse contexts, then the relevant object is not a static constructional label. It is a context-conditioned interpretation process.

This paper studies that problem through outcome-stable, force-variable utterances. In these cases, grammar constrains what kind of event is at issue, but discourse determines who does the act, who is responsible, and what social act the utterance is conveying. This view builds on structured approaches to linguistic interpretation, including constraint-based grammar and HPSG, where interpretation is not treated as a flat string label but as constrained, role-sensitive structure (Pollard and Sag, 1987, 1994; Sag et al., 2003). It also connects to speech-act and discourse-pragmatic traditions in which force, common ground, and conversational context are central to meaning (Austin, 1962; Searle, 1969; Grice, 1975; Stalnaker, 1978; Roberts, 2012; Ginzburg, 2012). For example, a directive causative such as *file khulvāo* can preserve the outcome of a file being opened while shifting the addressee from direct executor to coordinator. Similarly, an elliptical directive such as *roṭṭi ta* can preserve the outcome of rotis becoming available while shifting between instruction, delegation, and command. The form is stable; the force is not.

We formalize this contrast using an Outcome–Role–Attitude decomposition. For an utterance u in context c, we write interpretation as

$$I(c, u) = (O(u), R(c, u), A(c, u)),$$

where $O(u)$ is the outcome space licensed by the utterance form, $R(c, u)$ is the discourse-resolved role or responsibility assignment, and $A(c, u)$ is the attitude, stance, or interactional force. The central asymmetry is:

$$\phi(u) \Rightarrow O(u), \qquad \phi(u) \not\Rightarrow R(c, u), A(c, u).$$

That is, utterance-local form constrains outcome, but does not determine role or force.

This creates a fork for constructional accounts of LLM interpretation. If constructions are treated as local, discourse-invariant form–meaning pairings, they cannot explain cases where $\phi(u)$ is fixed but $I(c_1, u) \neq I(c_2, u)$. If constructions are extended to include discourse context, then the explanatory object is no longer a local construction alone, but a context-conditioned process. We take this fork as the starting point for interpretability. The question is not simply whether LLMs learn constructions. The question is what changes inside the model when the same constructional form receives different discourse-resolved force.

Our answer is that LLMs should be analyzed through bounded, context-conditioned interpretation trajectories. Transformer models compute finite layerwise hidden-state traces over input sequences (Vaswani et al., 2017; Devlin et al., 2019), and prior interpretability work has shown that linguistic and semantic information can be studied through layerwise representations, probing, and output-space readouts (Tenney et al., 2019; Rogers et al., 2020; Belinkov, 2022). For an input $x = (c, u)$, a transformer evolves a finite hidden-state trace,

$$H^{(0)}(x) \to H^{(1)}(x) \to \cdots \to H^{(L)}(x),$$

and each target-token state has an output-space readout

$$z_t^{(\ell)}(x) = h_t^{(\ell)}(x) W_U.$$

The resulting layerwise logit trajectory is not a separate object from the computation; it is the output-space trace of a bounded state transition. If role or force changes while form and outcome remain stable, successful interpretation should therefore produce ORA-relevant trajectory divergence.

We test this prediction empirically by constructing ORA contrast pairs and extracting layerwise hidden-state and logit trajectories from LLMs. Our results show that fixed-form, outcome-stable pairs produce divergent logit trajectories when role or force changes. Outcome-compatible probe mass remains comparatively stable, while role- and attitude-sensitive probe mass diverges across layers. Hidden-state trajectories show corresponding context-conditioned separation, supporting the view that ORA-sensitive interpretation is resolved through bounded state dynamics rather than static construction lookup.

The contribution is threefold. First, we identify a constructional fork for LLM interpretability: either local constructions fail on ORA-sensitive cases, or discourse extension turns interpretation into a process. Second, we introduce ORA as a compact linguistic target for separating outcome stability from role and force resolution. Third, we show that LLMs expose this resolution through layerwise logit and hidden-state trajectories. The resulting claim is not that transformers are classical symbolic machines. It is that, for discourse-sensitive interpretation, they behave as bounded state interpreters: finite, context-conditioned systems whose output-space trajectories reveal how meaning is resolved.

2 The Failure of Local Constructional Interpretation

Radical Construction Grammar makes constructions the basic units of grammar: form–meaning pairings from which categories and interpretation are derived (Croft, 2001). For LLM interpretability, this suggests a tempting claim: perhaps models learn constructions, and interpretation is recovered from utterance-local form. The ORA class breaks that claim. In these cases, the form $\phi(u)$ is fixed and the outcome space $O(u)$ is stable, but the interpretation changes because role and stance vary with discourse:

$$\begin{gathered} \phi(u) \text{ fixed}, \\ O(u) \text{ stable}, \\ I(c_1, u) \neq I(c_2, u). \end{gathered}$$

Thus no local form–meaning pairing can recover the full interpretation. If constructional locality is preserved, the account fails. If discourse is added to repair it, the object is no longer a local construction but a context-conditioned interpretation process. This is not an edge case: the pattern is productive and cross-linguistic across English, Hindi, Marathi, and Kannada directive, causative, and elliptical forms. These forms share stable outcome-spaces, underspecified execution, and discourse-dependent role/stance resolution. The full formal contradiction and construction-set precedence argument are given in Appendices E–F

3 ORA: Outcome, Role, and Stance

3.1 Outcome-Stable, Force-Variable Utterances

The target class is not defined by ambiguity alone. It is defined by a split between what form stabilizes

and what discourse resolves. In outcome-stable, force-variable utterances, the utterance-local form constrains the outcome space, but does not determine role assignment or stance. Thus, two uses of the same form may remain about the same broad event while changing who is responsible, who is targeted, or what social act is being performed.

A simple English example is *fuck you*. As a local form, it does not uniquely specify a literal action. Across contexts, it may function as an insult, dismissal, joking solidarity, defiance, sexual aggression, or threat. The form constrains a family of hostile, affective, or intimate force possibilities, but the actual stance is resolved only in discourse. The same issue appears in the South Asian cases we study: forms such as *file khulvāo* and *roṭṭi ta* stabilize an outcome while leaving role and stance open.

We call such cases *outcome-stable* because the utterance continues to anchor interpretation to a restricted event or force domain. We call them *force-variable* because the discourse can shift the utterance across instruction, request, delegation, insult, humiliation, dominance, or solidarity without requiring a different local form.

3.2 The ORA Decomposition

We represent interpretation as:

$$I(c, u) = (O(u), R(c, u), A(c, u)).$$

Here $O(u)$ is the outcome space licensed by utterance form, $R(c, u)$ is the discourse-resolved role assignment, and $A(c, u)$ is the stance or interactional force. The important asymmetry is:

$$\phi(u) \Rightarrow O(u), \qquad \phi(u) \not\Rightarrow R(c, u), A(c, u).$$

Form gives the model a constrained interpretive region; discourse resolves who occupies which role and what force the utterance carries.

ORA is not HPSG, and it is not an attribute-value matrix. HPSG gives a rich typed feature-structure account of grammar, with detailed constraints over syntax, valence, lexical information, and semantic roles. ORA does not attempt to reproduce that machinery. It is a reduced interpretability target: a projection of structured linguistic interpretation into three coordinates that can be tracked in model behavior and model internals. In this sense, ORA inherits the HPSG intuition that interpretation is structured and role-sensitive, but rejects the idea that we need a full grammar formalism to study what a transformer is resolving.

This matters because a model can appear to understand an utterance while failing one coordinate. It may preserve $O(u)$, knowing that *file khulvāo* concerns file opening, while misassigning $R(c, u)$, treating a manager as direct executor rather than coordinator. Or it may preserve the outcome and role while missing $A(c, u)$, flattening insult or humiliation into neutral instruction. ORA separates these cases. It defines what must remain stable, what must change with discourse, and what a logit trajectory should reveal if the model is resolving meaning rather than merely matching form.

4 Logit Trajectories as Bounded Interpretation

4.1 Layerwise Logit Trajectories

The constructional failure above gives a model-side question: if the relevant contrast is not recoverable from local form, where does the model resolve it? We treat the answer as trajectory-level. For an input $x = (c, u)$, where c is discourse context and u is the target utterance, a transformer defines a finite hidden-state trace

$$H^{(0)}(x) \to H^{(1)}(x) \to \cdots \to H^{(L)}(x),$$

with target-token states $h_t^{(\ell)}(x)$. Let $W_U \in \mathbb{R}^{d \times |V|}$ be the output projection. The layerwise logit readout is

$$z_t^{(\ell)}(x) = h_t^{(\ell)}(x) W_U,$$

and the logit trajectory is

$$T_t(x) = \left(z_t^{(0)}(x), z_t^{(1)}(x), \ldots, z_t^{(L)}(x) \right).$$

This trajectory is not independent of the hidden computation. It is the output-space readout of the model's finite state transition trace. In particular,

$$z_t^{(\ell+1)} - z_t^{(\ell)} = \left(h_t^{(\ell+1)} - h_t^{(\ell)} \right) W_U,$$

so every hidden-state displacement induces an exact output-space displacement. The full derivation, including the attention-stage contribution to logit movement, is given in Appendix I.

4.2 ORA Probe Trajectories

To connect logit trajectories to ORA, we define diagnostic probe sets over the vocabulary:

$$\Omega_O(u) \subseteq V, \qquad \Omega_R(c, u) \subseteq V, \qquad \Omega_A(c, u) \subseteq V.$$

These correspond to outcome-compatible, role-compatible, and stance-compatible continuation families. For any logit vector z, the probe mass of a set Ω is

$$M_\Omega(z) = \sum_{i \in \Omega} \mathrm{softmax}(z)_i.$$

The ORA-interpreted state at layer ℓ is then

$$\begin{aligned} m_t^{(\ell)}(c,u) = \big(& M_{\Omega_O(u)}(z_t^{(\ell)}), \\ & M_{\Omega_R(c,u)}(z_t^{(\ell)}), \\ & M_{\Omega_A(c,u)}(z_t^{(\ell)}) \big). \end{aligned}$$

and the interpreted trajectory is

$$\begin{aligned} \widetilde{T}_t(c,u) = \big(& m_t^{(0)}(c,u), m_t^{(1)}(c,u), \\ & \ldots, m_t^{(L)}(c,u) \big). \end{aligned}$$

This gives a compact readout of how the model's output-relevant state moves with respect to outcome, role, and stance across layers.

4.3 Bounded Trajectory Divergence

The central prediction is that ORA-sensitive interpretation should appear as trajectory divergence. Let (c_1, u) and (c_2, u) be an outcome-stable pair:

$$O(c_1, u) = O(c_2, u) = O(u),$$

but suppose role or stance differs:

$$R(c_1, u) \neq R(c_2, u) \quad \text{or} \quad A(c_1, u) \neq A(c_2, u).$$

If a transformer successfully resolves both inputs, then the two interpreted trajectories cannot be identical:

$$\widetilde{T}_t(c_1, u) \neq \widetilde{T}_t(c_2, u).$$

Equivalently, there must exist some layer ℓ and some ORA-relevant probe set

$$\Omega \in \{\Omega_R, \Omega_A\}$$

such that

$$M_\Omega\Big(z_t^{(\ell)}(c_1, u)\Big) \neq M_\Omega\Big(z_t^{(\ell)}(c_2, u)\Big).$$

This is the bounded interpretation claim. A transformer that resolves ORA does not simply map an utterance form to a constructional label. It evolves a finite context-conditioned state trajectory, and the logit trajectory is the output-space trace of that computation. The main empirical question is therefore whether fixed-form, outcome-stable pairs produce role- and stance-sensitive trajectory divergence in actual LLMs.

5 Empirical Evidence for Bounded ORA Interpretation

5.1 Experimental Setup

We run four empirical probes of the framework. The first two are ORA-specific: they test whether fixed-form, outcome-stable utterance pairs produce layerwise role/stance divergence in output space. The third tests whether transformer hidden states can encode and update a bounded latent state. The fourth tests whether prompt-conditioned hidden-state transitions have regime-sensitive operator structure beyond the ORA examples.

For ORA, each item is a pair $(c_1, u), (c_2, u)$ with the same surface utterance u, different contexts, stable outcome probes, and different role or stance probes. We use `distilgpt2` and extract hidden states at the final prompt position from every layer. Each hidden state is projected through the LM head:

$$z_t^{(\ell)}(c,u) = h_t^{(\ell)}(c,u) W_U.$$

Probe words are mapped to first-token ids. For each ORA coordinate, probe mass is computed by a local softmax over the competing probes in that coordinate:

$$M_\Omega(z) = \sum_{i \in \Omega} \frac{\exp z_i}{\sum_{j \in G(\Omega)} \exp z_j},$$

where $G(\Omega)$ is the union of competing probes for that coordinate. Thus role probes compete with role probes, and stance probes compete with stance probes. This avoids treating ORA as a global vocabulary prediction task.

For each layer, we compute role divergence, stance divergence, and combined role–stance divergence:

$$D_R^{(\ell)} = \left\| \begin{bmatrix} M_{R_1}(z_{c_1}^{(\ell)}) \\ M_{R_2}(z_{c_1}^{(\ell)}) \end{bmatrix} - \begin{bmatrix} M_{R_1}(z_{c_2}^{(\ell)}) \\ M_{R_2}(z_{c_2}^{(\ell)}) \end{bmatrix} \right\|_2,$$

with $D_A^{(\ell)}$ defined analogously and $D_{RA}^{(\ell)}$ computed over all four role/stance coordinates. We summarize trajectories using final divergence, area under the divergence curve, curvature, and bifurcation depth, where bifurcation is the first layer for which $D_{RA}^{(\ell)} > 0.35$. Full prompts, probe sets, raw logits, and all curves are given in the appendix.

5.2 Logit Trajectory Divergence

The first result is that fixed-form ORA contrasts produce nonzero role/stance logit-trajectory divergence. In the hand-specified ORA set, every item

keeps the surface utterance fixed across contexts. The model receives the same target form, but its layerwise probe trajectory changes when the discourse changes.

The strongest cases are affective force shifts: *damn you* and *fuck you* show large role/stance divergence because the same string moves between affiliative teasing and hostile rejection. The Indic and institutional items show smaller but still nonzero divergence. We do not treat this as a failure of the theory; with `distilgpt2`, the Indic surface forms are weakly represented, and much of the signal enters through English context and probe labels. The result is therefore conservative: even a small English-heavy model shows context-conditioned trajectory movement under fixed form.

5.3 Outcome Persistence and Role/Stance Divergence

The second result is that the divergence is structured, not arbitrary. ORA predicts that the outcome coordinate should remain comparatively stable while role and stance separate. To test this beyond the hand-specified set, we run a candidate sweep over affective, institutional, and Indic/mixed forms. Each candidate has the same surface form across contexts, stable outcome probes, and competing role/stance probes. We rank candidates by role–stance divergence after accounting for outcome drift.

The top cases are exactly the cases where discourse changes the force while preserving the form. Affective examples have the largest mean D_{RA} AUC. Institutional examples are lower but still visible when the same directive shifts from execution to authorization, as in *clear the case*. Five of seventeen sweep candidates cross the bifurcation threshold. This matters because the effect is not only a final-output artifact: the role/stance separation emerges along the layerwise trajectory.

5.4 Hidden-State Trajectory Divergence

The ORA results show bounded interpretation in output-space readouts. We next test whether hidden states themselves can carry bounded latent state. For this, we use a controlled parenthesis-depth task. We generate 700 bounded sequences over "(" and ")", with depth clipped at 6. The true latent state is:

$$d_t = \#\text{ unmatched left parentheses after token } t.$$

At each parenthesis token, we extract hidden states from every layer and train a linear probe to decode d_t. We then test transition consistency: if the next symbol is "(", decoded depth should increase by one; if the next symbol is ")", it should decrease by one, bounded to $[0, 6]$.

The embedding layer is weak, but the first transformer layers rapidly organize the latent state. Depth accuracy rises from 0.466 at layer 0 to 0.931 at layer 1, and peaks at 0.998 at layer 3. Transition consistency follows the same pattern, peaking at 0.994. This is direct hidden-state evidence for bounded state tracking: the model's intermediate states linearly encode a finite latent variable and preserve its update dynamics.

Utterance	$\mathbf{D_R}$	$\mathbf{D_A}$	$\mathbf{D_{RA}}$
damn you	1.088	0.047	1.089
fuck you	0.607	0.254	0.658
khulvane-ka sikh	0.288	0.033	0.290
motherfucking bastard	0.174	0.122	0.213
rotti ta	0.110	0.162	0.196
goddamnit	0.175	0.065	0.186
chavi kadh	0.095	0.021	0.097
file khulvao	0.083	0.005	0.084

Table 1: Role, stance, and combined role–stance trajectory distances for hand-specified ORA pairs. Each row compares two discourse contexts with the same surface utterance.

Utterance	Class	$\mathrm{AUC}(\mathbf{D_{RA}})$	Bif. Layer
fuck you	affective	1.405	5
clear the case	institutional	1.360	2
you monster	affective	1.069	5
go to hell	affective	1.021	5
damn you	affective	0.787	–
you idiot	affective	0.783	5

Table 2: Top ORA trajectory-geometry cases from the candidate sweep. AUC is computed over layerwise combined role/stance divergence. Bifurcation layer is the first layer where $D_{RA}^{(\ell)} > 0.35$; "–" indicates no threshold crossing.

5.5 Regime-Specific Prompt–Response Operators

The final empirical block asks whether bounded trajectory structure appears beyond hand-built ORA pairs and synthetic automata. We analyze 5000 prompt–response pairs extracted from conversation data. Each pair is labeled by prompt task and response family. For every example, we extract hidden states at two positions: the end of the prompt and the end of the response. This gives, for each layer,

$$h^{(\ell)}_{\text{prompt}}, \qquad h^{(\ell)}_{\text{response}}.$$

Layer	Depth Acc.	Depth MAE	Transition Acc.
0	0.466	1.378	0.530
1	0.931	0.069	0.900
2	0.951	0.049	0.903
3	0.998	0.002	0.994
4	0.975	0.025	0.963
5	0.970	0.030	0.954
6	0.953	0.047	0.938

Table 3: Bounded-state tracking in hidden states. Linear probes decode parenthesis depth from intermediate layers, and decoded states obey the expected bounded transition rule.

Regime	Type	Best Layer	Best Error
clarification	response family	6	0.562
code	prompt task	5	0.573
writing	prompt task	4	0.656
answer	response family	5	0.699
GLOBAL	global	5	0.736
code	response family	5	0.742
general	prompt task	3	0.833

Table 4: Rank-32 PCA prompt–response operator fits. Several task- or response-conditioned operators outperform the global operator, indicating regime-sensitive hidden-state transition structure.

We then fit linear transition operators from prompt-end state to response-end state:

$$h^{(\ell)}_{\text{response}} \approx A^{(\ell)}_{r} h^{(\ell)}_{\text{prompt}},$$

where r may be global, prompt-task-specific, or response-family-specific. We evaluate relative reconstruction error. We also repeat the fit in rank-32 PCA subspaces to test whether low-dimensional regime operators capture the transition.

This experiment matters because it tests the broader bounded interpretation claim. If hidden trajectories were unstructured drift, prompt–response transitions should not admit stable low-rank operator descriptions. Instead, several regimes show lower error than the global operator. In rank-32 PCA space, the global operator reaches best relative error 0.736. Regime-specific operators improve on this: clarification responses reach 0.562, code prompts 0.573, writing prompts 0.656, and answer responses 0.699.

The same feature extraction also supports linear decoding of prompt task and response family. Prompt task is decodable from prompt-end hidden states with test accuracy up to 0.868, while response family is decodable from response-end hidden states with test accuracy up to 0.819. These results do not claim that the model explicitly stores symbolic regimes. They show something weaker and more useful: hidden-state trajectories contain regime-sensitive structure that can be read out and approximated by low-rank operators.

Taken together, the four empirical probes support the same interpretation. ORA examples show that fixed forms induce different role/stance logit trajectories under different contexts. ORA geometry shows that this divergence unfolds across layers. The automata probe shows that hidden states can encode and update bounded latent state. Regime operators show that prompt-conditioned hidden transitions are structured by task and response family. The empirical evidence therefore supports the bounded interpretation view: LLMs do not merely retrieve local constructional labels; they evolve finite, context-conditioned trajectories whose readouts carry interpretable state information.

6 Discussion

The results support a different unit of analysis for LLM interpretation. The relevant object is not a static constructional label, but a bounded trajectory by which the model resolves an utterance in context. ORA cases make this visible because they hold form relatively fixed while changing role and stance. If a model were only recovering local constructional form, these cases should collapse. Instead, the output-space trajectories separate across contexts, and they separate in the coordinates predicted by the linguistic decomposition.

This does not mean that constructions are irrelevant. The opposite is closer to the claim: form matters because it constrains the outcome space. In ORA terms, constructional form can stabilize $O(u)$. The failure is not that constructional information disappears, but that it is insufficient. Role and stance are not fully present in the local form; they are resolved through discourse. This is why the constructional fork matters. A local constructional account cannot recover $I(c, u)$ for the target class. A discourse-extended constructional account can recover the contrast, but only by shifting the explanatory object from a local form–meaning pairing to a context-conditioned interpretation process.

The logit trajectory view makes that process measurable. A transformer evolves a finite hidden-state trace, and layerwise logits provide an output-space readout of that trace. ORA probe trajectories then allow us to ask whether the model's state is moving in outcome-, role-, or stance-relevant direc-

tions. The empirical results show that fixed-form contrasts produce role/stance-sensitive logit divergence, while the automata and regime-operator experiments show that hidden states can encode bounded latent variables and task-conditioned transition structure. These are different empirical containers, but they point to the same conclusion: transformer interpretation is better understood as bounded state dynamics with readout than as construction lookup.

This also narrows the paper's claim. We do not claim that LLMs explicitly represent O, R, or A as symbolic variables. We do not claim that probe masses are the model's internal semantics. We also do not claim that attention weights are explanations. The claim is weaker and more useful: for a productive class of utterances where form does not determine interpretation, successful computation must depend on context-sensitive internal trajectories, and those trajectories leave measurable traces in hidden states and output-space readouts.

The broader implication is methodological. Interpretability needs linguistic targets that are neither full grammatical theories nor flat task labels. HPSG and related constraint-based traditions show why interpretation should be structured and role-sensitive; Construction Grammar shows why form matters; discourse pragmatics shows why force cannot be ignored. ORA compresses these insights into a target that can be operationalized inside transformer analysis. It gives us a way to ask not simply whether a model "understands" an utterance, but which part of interpretation it stabilizes, which part it resolves, and where that resolution appears along the computation.

7 Related Work

Constructional and constraint-based meaning. Construction Grammar treats linguistic knowledge as organized around form–meaning pairings, from early constructional analyses to argument-structure and usage-based accounts (Fillmore et al., 1988; Goldberg, 1995, 2006; Kay and Fillmore, 1999; Diessel, 2019). Radical Construction Grammar pushes this further by treating constructions as primitive units from which categories are derived (Croft, 2001). Constraint-based grammar and HPSG instead emphasize typed feature structures, lexical constraints, valence, and role-sensitive interpretation (Pollard and Sag, 1987, 1994; Sag et al., 2003; Ginzburg and Sag, 2000). ORA is not a replacement for these frameworks; it is a reduced interpretability target motivated by their shared insight that meaning is structured rather than a flat label.

Discourse, pragmatics, and force. The role/stance side of ORA connects to speech-act theory, implicature, common ground, discourse structure, and interactional meaning (Austin, 1962; Searle, 1969; Grice, 1975; Levinson, 1983; Stalnaker, 1978; Kamp and Reyle, 1993; Asher and Lascarides, 2003; Roberts, 2012; Ginzburg, 2012). Our claim differs from treating pragmatics as optional enrichment. In the target class, discourse is required to identify role and stance; it is not merely an afterthought added to a complete local constructional meaning.

LLMs, constructions, and interpretability. Recent work has begun to test whether language models encode or generalize constructional information (Weissweiler et al., 2023; Tayyar Madabushi et al., 2023; Xu et al., 2023; Scivetti et al., 2025). A separate line studies what linguistic information is available in transformer representations through probing, layerwise analysis, attention analysis, and broader BERTology-style interpretability (Vaswani et al., 2017; Devlin et al., 2019; Tenney et al., 2019; Rogers et al., 2020; Belinkov, 2022). Logit-lens and related output-space methods motivate our use of layerwise vocabulary readouts as trajectories rather than final-layer-only predictions (nostalgebraist, 2020; Belrose et al., 2023; Geva et al., 2021, 2022). Our contribution is to connect these methods to a specific linguistic failure mode for local constructional interpretation.

8 Conclusion

The same form can carry different force. For outcome-stable ORA utterances, local form constrains the outcome space, but discourse resolves role and stance. This creates a failure mode for local constructional interpretation and a target for LLM interpretability.

We argued that LLMs should be analyzed not only as learners of constructions, but as bounded state interpreters: finite, context-conditioned systems whose hidden-state trajectories and logit readouts resolve discourse-sensitive meaning. Empirically, fixed-form ORA contrasts produce role/stance-sensitive logit divergence; bounded automata states are recoverable from hidden trajecto-

ries; and prompt–response transitions show regime-sensitive operator structure. Together, these results suggest that what LLMs learn is not merely a static inventory of form–meaning pairings, but a family of context-conditioned trajectories by which meaning is resolved.

Limitations

This paper focuses on a specific but productive class of utterances where outcome remains stable while role and stance vary with discourse. ORA is intended as a compact interpretability target rather than a complete theory of meaning, and the empirical results should be read as evidence for the proposed trajectory-based framework rather than as an exhaustive benchmark across all models and languages. Future work should extend the analysis to larger multilingual and instruction-tuned models, native-script examples, richer probe sets, and human-validated annotations.

Ethics Statement

This work analyzes model behavior and representation trajectories. Some examples include profanity or emotionally charged utterances because they expose clear shifts in stance across context; they are used only for linguistic and interpretability analysis. One empirical component uses an internal corpus of LLM conversation pairs for aggregate prompt–response trajectory analysis. We do not quote, reproduce, or release private conversations; the paper reports only derived labels, hidden-state features, probe scores, and aggregate metrics. Any public release should include only code, synthetic examples, non-identifying plots, and aggregate results.

References

Nicholas Asher and Alex Lascarides. 2003. *Logics of Conversation*. Cambridge University Press, Cambridge.

J. L. Austin. 1962. *How to Do Things with Words*. Oxford University Press, Oxford.

Yonatan Belinkov. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219.

Nora Belrose, Igor Ostrovsky, Lev McKinney, Zach Furman, Logan Smith, Danny Halawi, Stella Biderman, and Jacob Steinhardt. 2023. Eliciting latent predictions from transformers with the tuned lens. *Preprint*, arXiv:2303.08112.

William Croft. 2001. *Radical Construction Grammar: Syntactic Theory in Typological Perspective*. Oxford University Press, Oxford.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Holger Diessel. 2019. *The Grammar Network: How Linguistic Structure Is Shaped by Language Use*. Cambridge University Press, Cambridge.

Charles J. Fillmore, Paul Kay, and Mary Catherine O'Connor. 1988. Regularity and idiomaticity in grammatical constructions: The case of let alone. *Language*, 64(3):501–538.

Mor Geva, Avi Caciularu, Kevin Wang, and Yoav Goldberg. 2022. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 30–45, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jonathan Ginzburg. 2012. *The Interactive Stance: Meaning for Conversation*. Oxford University Press, Oxford.

Jonathan Ginzburg and Ivan A. Sag. 2000. *Interrogative Investigations: The Form, Meaning, and Use of English Interrogatives*. CSLI Publications, Stanford, CA.

Adele E. Goldberg. 1995. *Constructions: A Construction Grammar Approach to Argument Structure*. University of Chicago Press, Chicago.

Adele E. Goldberg. 2006. *Constructions at Work: The Nature of Generalization in Language*. Oxford University Press, Oxford.

H. Paul Grice. 1975. Logic and conversation. In Peter Cole and Jerry L. Morgan, editors, *Syntax and Semantics 3: Speech Acts*, pages 41–58. Academic Press, New York.

Hans Kamp and Uwe Reyle. 1993. *From Discourse to Logic*. Kluwer Academic Publishers, Dordrecht.

Paul Kay and Charles J. Fillmore. 1999. Grammatical constructions and linguistic generalizations: The what's x doing y? construction. *Language*, 75(1):1–33.

Stephen C. Levinson. 1983. *Pragmatics*. Cambridge University Press, Cambridge.

nostalgebraist. 2020. Interpreting GPT: The logit lens. LessWrong.

Carl Pollard and Ivan A. Sag. 1987. *Information-Based Syntax and Semantics, Volume 1: Fundamentals*. CSLI Publications, Stanford, CA.

Carl Pollard and Ivan A. Sag. 1994. *Head-Driven Phrase Structure Grammar*. University of Chicago Press, Chicago.

Craige Roberts. 2012. Information structure in discourse: Towards an integrated formal theory of pragmatics. *Semantics and Pragmatics*, 5(6):1–69.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Ivan A. Sag, Thomas Wasow, and Emily M. Bender. 2003. *Syntactic Theory: A Formal Introduction*, 2 edition. CSLI Publications, Stanford, CA.

Wesley Scivetti, Melissa Torgbi, Austin Blodgett, Molly Shichman, Taylor Hudson, Claire Bonial, and Harish Tayyar Madabushi. 2025. Assessing language comprehension in large language models using construction grammar. *Preprint*, arXiv:2501.04661.

John R. Searle. 1969. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press, Cambridge.

Robert C. Stalnaker. 1978. Assertion. In Peter Cole, editor, *Syntax and Semantics 9: Pragmatics*, pages 315–332. Academic Press, New York.

Harish Tayyar Madabushi, Laurence Romain, Petar Milin, and Dagmar Divjak. 2023. Construction grammar and language models. *Preprint*, arXiv:2308.13315.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proceedings of ACL*, pages 4593–4601.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

Leonie Weissweiler, Taiqi He, Naoki Otani, David R. Mortensen, Lori Levin, and Hinrich Schütze. 2023. Construction grammar provides unique insight into neural language models. In *Proceedings of the First International Workshop on Construction Grammars and NLP*, pages 85–95. Association for Computational Linguistics.

Lvxiaowei Xu, Jianwang Wu, Jiawei Peng, Zhilin Gong, Ming Cai, and Tianxiang Wang. 2023. Enhancing language representation with constructional information for natural language understanding. *Preprint*, arXiv:2306.02819.

A Appendix Organization

The appendix mirrors the structure of the main paper. Appendices B–G give the linguistic object and constructional-failure proofs. Appendices H–O give the transformer, logit-trajectory, ORA-trajectory, and bounded-state formalization. Appendices Q–V give the empirical methods and result descriptions. Appendix W contains the complete tables, Appendix X contains the complete figures, and Appendices Y–Z list artifacts and controls.

For reproducibility, the artifact bundle contains CSV outputs, plots, and analysis scripts. In the camera-ready version, these artifacts should be linked through the project repository: `https://github.com/anonymous/ora-bounded-interpretation`.

B Full ORA Example Inventory

This section gives the linguistic inventory behind the ORA analysis. The main paper includes only representative examples; the full inventory is reported in Table 5. The purpose of the inventory is not to claim that each utterance has exactly two readings. Rather, it shows the pattern used throughout the paper: utterance-local form constrains an outcome domain, while role and stance are resolved in discourse.

The invariants are: (i) the utterance-local form $\phi(u)$ is fixed across discourse contexts; (ii) the outcome space $O(u)$ is constrained by form; (iii) role assignment $R(c, u)$ varies with discourse; and (iv) stance $A(c, u)$ varies with discourse.

We define the utterance-local form as information recoverable from the utterance without discourse context:

$$\phi(u) = \left\{ \begin{array}{l} \text{lexical material, morphology,} \\ \text{syntax, local composition} \end{array} \right\}.$$

For example:

$$\phi(\textit{file khulvāo}) = \left\{ \begin{array}{l} \textit{file, open, causative,} \\ \textit{imperative/directive} \end{array} \right\}.$$

This local form does not include who is speaking, what happened before, who has institutional authority, or what social act is being performed.

The outcome space is the restricted space of possible results licensed by form:

$$O(u) = \left\{ \begin{array}{l} \text{possible outcomes licensed} \\ \text{by } \phi(u) \end{array} \right\}.$$

For example:

$$O(\textit{file khulvāo}) = \left\{ \begin{array}{l} \text{file becomes open,} \\ \text{file is processed/opened} \end{array} \right\}.$$

This is constrained, but it is not a complete interpretation.

C ORA Annotation Guidelines

Each utterance-in-context pair (c, u) is annotated as:

$$I(c, u) = (O(u), R(c, u), A(c, u)).$$

Outcome $O(u)$. The outcome is the restricted event, result, or force domain licensed by utterance form. It is annotated from the utterance-local form rather than from the full discourse. The outcome may be broad. For example, *fuck you* does not encode a literal action; it licenses a direct affective address domain. *file khulvāo* licenses a file-opening or file-processing domain.

Role $R(c, u)$. The role coordinate records participant and responsibility assignment. It answers: who acts, who is affected, who is responsible, who coordinates, who authorizes, and who is targeted. For example, in *file khulvāo*, a clerk context assigns the addressee as executor, while a manager context assigns the addressee as coordinator or authority.

Stance $A(c, u)$. The stance coordinate records interactional force: instruction, request, command, delegation, humiliation, dominance, joking solidarity, rejection, threat, or insult. Stance is not simply sentiment. A joking insult and a hostile insult can share lexical polarity while differing in social force.

Annotation rule. Annotate $O(u)$ first from form. Then annotate $R(c, u)$ and $A(c, u)$ from context. If the context supports multiple readings, annotate the compatible set and mark the contrastive dimension. For ORA pairs, the intended contrast should preserve $O(u)$ while changing $R(c, u)$, $A(c, u)$, or both.

D Identifiability of Outcome, Role, and Stance

Theorem A.1. For the target utterance class, the interpretation $I(c, u)$ is not uniquely recoverable from any proper subset of $\{O, R, A\}$.

Proof. We isolate the failure caused by removing each coordinate.

Removing R. Assume that

$$I'(c, u) = (O(u), A(c, u))$$

is sufficient for interpretation. Consider *khulvāne-kā sīkh*. Two readings may share an opening outcome and an evaluative/instructional frame, but differ in role:

$$\begin{aligned} R_1 &= \text{addressee as agent,} \\ R_2 &= \text{addressee as incompetent target.} \end{aligned}$$

Since the two readings share O and A but differ in R, the pair (O, A) cannot identify the interpretation. Contradiction.

Removing A. Assume that

$$I'(c, u) = (O(u), R(c, u))$$

is sufficient. Consider two readings where the outcome is opening and the addressee is involved as the relevant participant, but the stance differs:

$$\begin{aligned} A_1 &= \text{neutral instruction,} \\ A_2 &= \text{dominance / threatened pedagogy.} \end{aligned}$$

The same O and R do not determine whether the utterance is neutral instruction or humiliative dominance. Contradiction.

Removing O. Assume that

$$I'(c, u) = (R(c, u), A(c, u))$$

is sufficient. Consider two readings where role and stance are both instructional/evaluative, but the outcome differs:

$$\begin{aligned} O_1 &= \text{bodily opening,} \\ O_2 &= \text{desire shift / abstract opening frame.} \end{aligned}$$

The same R and A do not determine the event or outcome space. Contradiction. Since removing any one of O, R, or A destroys identifiability, full ORA interpretation requires all three coordinates:

$$I(c, u) = (O(u), R(c, u), A(c, u)).$$

E Local Construction Insufficiency

Let u be a target utterance, c a discourse context, and $\phi(u)$ the utterance-local representation of u. Let $Interp(c, u) \in Y$ be the correct interpretation.

Theorem B.1. If there exist contexts c_1, c_2 such that $\phi(u)$ is fixed and

$$Interp(c_1, u) \neq Interp(c_2, u),$$

then there does not exist a discourse-invariant local construction function

$$f_{lc} : \phi \to Y$$

such that

$$Interp(c, u) = f_{lc}(\phi(u))$$

for both contexts.

Proof. Assume for contradiction that such a function f_{lc} exists. Since f_{lc} is discourse-invariant, it depends only on $\phi(u)$. For a discourse-contrast pair $(c_1, u), (c_2, u)$, the utterance-local form is identical:

$$\phi_{c_1}(u) = \phi_{c_2}(u) = \phi(u).$$

Therefore:

$$f_{lc}(\phi_{c_1}(u)) = f_{lc}(\phi_{c_2}(u)).$$

Hence:

$$Interp(c_1, u) = Interp(c_2, u).$$

But this contradicts:

$$Interp(c_1, u) \neq Interp(c_2, u).$$

Therefore no such discourse-invariant local-construction function exists.

Instantiation. Let:

$$u = \textit{khulvāne-kā sīkh}.$$

Let c_1 be a humiliative technique-instruction context and c_2 a general competence-attack context. In both contexts:

$$\phi(u) = \left\{ \begin{array}{l} \text{open, causative, learn,} \\ \text{directive/evaluative frame} \end{array} \right\}.$$

But:

$$Interp(c_1, u) \neq Interp(c_2, u),$$

because c_1 resolves the utterance as an instruction/degrading pedagogy, while c_2 resolves it as a competence attack or dominance move. Thus a local constructional mapping cannot recover both readings.

F Construction Set and Fixed Precedence Insufficiency

A possible repair is to allow the utterance to license multiple constructions. We show that this does not solve the problem if the construction set and precedence rule remain discourse-invariant.

Let:

$$C(u) = \{C_1, \ldots, C_n\}$$

be the set of constructions licensed by utterance-local form. Let:

$$\pi : C(u) \to \mathbb{R}$$

be a fixed precedence or ranking function. Suppose interpretation is determined by:

$$Interp(c, u) = G\left(\arg\max_{C_i \in C(u)} \pi(C_i)\right).$$

Theorem B.2. If $C(u)$ and π are discourse-invariant, then no such construction-set-plus-precedence account can recover a discourse-contrast pair with fixed u and different interpretations.

Proof. Assume $C(u)$ depends only on u, and π is fixed across contexts. Then for (c_1, u) and (c_2, u), the same candidate set is available:

$$C_{c_1}(u) = C_{c_2}(u) = C(u).$$

The same ranking function applies:

$$\pi_{c_1} = \pi_{c_2} = \pi.$$

Therefore, the same winning construction is selected in both contexts:

$$\arg\max_{C_i \in C(u)} \pi(C_i).$$

Since G is also fixed, the resulting interpretation is identical:

$$Interp(c_1, u) = Interp(c_2, u).$$

This contradicts:

$$Interp(c_1, u) \neq Interp(c_2, u).$$

Therefore, a discourse-invariant construction set with fixed precedence is insufficient.

Consequence. Either constructional locality is preserved, in which case the theory fails for this utterance class, or discourse is incorporated into construction selection, in which case the explanatory object is no longer a local form–meaning pairing. It becomes a context-conditioned interpretation process.

G Why This Is Not Pragmatic Add-On

The ORA phenomenon is not ordinary pragmatic enrichment. In standard enrichment, a constructional or compositional meaning is selected first, and context adds nuance. In ORA cases, discourse is required to resolve the interpretive object itself. Before discourse resolution, we may know the outcome domain, but not the role assignment or stance.

The relevant properties are: (i) grammar underdetermines role and stance; (ii) multiple incompatible role/stance resolutions are licensed by the same local form; (iii) no unique constructional meaning is selected before discourse resolution; and (iv) the class is productive rather than accidental.

The class is productive across English, Hindi, Marathi, Kannada, and related contact forms. Examples include *khulvāo*, *chāvi kāḍ*, *khulvāne-kā sīkh*, *bhākrī ān*, *roṭṭi ta*, and related directive/causative/elliptical forms. They share stable outcome spaces, underspecified execution, and discourse-dependent interpretation. Thus this is not an edge case; it is an extensible class of grammatical forms.

H Transformer Setup and Attention Dynamics

Let the vocabulary be:

$$V = \{1, \ldots, N\}.$$

Let the input sequence be:

$$x = (x_1, \ldots, x_T), \qquad x_i \in V.$$

For ORA utterances, decompose:

$$x = (c, u),$$

where:

$$c = (x_1, \ldots, x_m)$$

is the discourse context and:

$$u = (x_{m+1}, \ldots, x_T)$$

is the target utterance. Let $t \in \{m+1, \ldots, T\}$ be a token position inside u.

A transformer with L layers defines hidden states:

$$h_t^{(0)}, h_t^{(1)}, \ldots, h_t^{(L)} \in \mathbb{R}^d.$$

The full hidden-state matrix at layer ℓ is:

$$H^{(\ell)}(x) = \left(h_1^{(\ell)}, \ldots, h_T^{(\ell)}\right) \in \mathbb{R}^{T \times d}.$$

The input embedding is:

$$h_i^{(0)} = E[x_i] + p_i.$$

Each layer is a map:

$$F_\theta^{(\ell)} : \mathbb{R}^{T \times d} \to \mathbb{R}^{T \times d}$$

such that:

$$H^{(\ell+1)}(x) = F_\theta^{(\ell)}(H^{(\ell)}(x)).$$

For attention head a at layer ℓ, define:

$$q_t^{(\ell,a)} = h_t^{(\ell)} W_Q^{(\ell,a)},$$
$$k_j^{(\ell,a)} = h_j^{(\ell)} W_K^{(\ell,a)},$$
$$v_j^{(\ell,a)} = h_j^{(\ell)} W_V^{(\ell,a)}.$$

The causal attention weights are:

$$\alpha_{t,j}^{(\ell,a)} = \frac{\exp\left(q_t^{(\ell,a)} \cdot k_j^{(\ell,a)}/\sqrt{d_k}\right)}{\sum_{r\le t}\exp\left(q_t^{(\ell,a)} \cdot k_r^{(\ell,a)}/\sqrt{d_k}\right)}, \quad j \le t.$$

The head output is:

$$o_t^{(\ell,a)} = \sum_{j\le t} \alpha_{t,j}^{(\ell,a)} v_j^{(\ell,a)}.$$

The multi-head attention output is:

$$Attn_t^{(\ell)} = Concat(o_t^{(\ell,1)}, \ldots, o_t^{(\ell,H)}) W_O^{(\ell)}.$$

With residual connection and layer normalization:

$$\bar{h}_t^{(\ell)} = LN\left(h_t^{(\ell)} + Attn_t^{(\ell)}\right),$$

and:

$$h_t^{(\ell+1)} = LN\left(\bar{h}_t^{(\ell)} + FFN^{(\ell)}(\bar{h}_t^{(\ell)})\right).$$

I Layerwise Logits and Exact Logit Recursion

Let the unembedding/output projection be:

$$W_U \in \mathbb{R}^{d\times N}.$$

Define layerwise logits:

$$z_t^{(\ell)}(x) = h_t^{(\ell)}(x) W_U \in \mathbb{R}^N.$$

The layerwise predictive distribution is:

$$p_t^{(\ell)}(x) = softmax(z_t^{(\ell)}(x)).$$

The logit trajectory at target token t is:

$$T_t(x) = \left(z_t^{(0)}(x), z_t^{(1)}(x), \ldots, z_t^{(L)}(x)\right).$$

Theorem C.1: Exact Logit Recursion. For every layer ℓ and target token position t,

$$z_t^{(\ell+1)} - z_t^{(\ell)} = \left(h_t^{(\ell+1)} - h_t^{(\ell)}\right) W_U.$$

Equivalently:

$$z_t^{(\ell+1)} = z_t^{(\ell)} + \left(h_t^{(\ell+1)} - h_t^{(\ell)}\right) W_U.$$

Proof. By definition:

$$z_t^{(\ell)} = h_t^{(\ell)} W_U$$

and:

$$z_t^{(\ell+1)} = h_t^{(\ell+1)} W_U.$$

Subtracting gives:

$$\begin{aligned} z_t^{(\ell+1)} - z_t^{(\ell)} &= h_t^{(\ell+1)} W_U - h_t^{(\ell)} W_U \\ &= \left(h_t^{(\ell+1)} - h_t^{(\ell)}\right) W_U. \end{aligned}$$

Thus the logit trajectory is not an independent object. It is the image of the hidden-state trajectory under the fixed output projection W_U.

J Exact Attention Contribution to Logit Update

Define the attention-stage logits:

$$\bar{z}_t^{(\ell)} = \bar{h}_t^{(\ell)} W_U.$$

Theorem C.2: Attention-Stage Logit Displacement. At the attention stage,

$$\bar{z}_t^{(\ell)} - z_t^{(\ell)} = \left(\bar{h}_t^{(\ell)} - h_t^{(\ell)}\right) W_U.$$

If layer normalization is replaced by the identity map, then:

$$\bar{z}_t^{(\ell)} - z_t^{(\ell)} = Attn_t^{(\ell)} W_U.$$

Proof. By definition:

$$\bar{z}_t^{(\ell)} = \bar{h}_t^{(\ell)} W_U$$

and:

$$z_t^{(\ell)} = h_t^{(\ell)} W_U.$$

Therefore:

$$\bar{z}_t^{(\ell)} - z_t^{(\ell)} = \left(\bar{h}_t^{(\ell)} - h_t^{(\ell)}\right) W_U.$$

If LN is the identity, then:

$$\bar{h}_t^{(\ell)} = h_t^{(\ell)} + Attn_t^{(\ell)}.$$

So:

$$\bar{h}_t^{(\ell)} - h_t^{(\ell)} = Attn_t^{(\ell)}.$$

Substitution yields:

$$\bar{z}_t^{(\ell)} - z_t^{(\ell)} = Attn_t^{(\ell)} W_U.$$

To avoid a wide expansion, define:

$$a_t^{(\ell,h)} = \sum_{j\le t} \alpha_{t,j}^{(\ell,h)} v_j^{(\ell,h)}.$$

Then:

$$Attn_t^{(\ell)} = Concat(a_t^{(\ell,1)}, \ldots, a_t^{(\ell,H)}) W_O^{(\ell)}.$$

Thus attention-stage logit movement is generated by a weighted contextual mixture of value vectors, projected through output space.

K ORA Probe Masses and Interpreted Trajectories

For an ORA utterance:

$$I(c,u) = (O(u), R(c,u), A(c,u)).$$

Define diagnostic probe sets:

$$\Omega_O(u) \subseteq V, \quad \Omega_R(c,u) \subseteq V, \quad \Omega_A(c,u) \subseteq V.$$

For any logit vector $z \in \mathbb{R}^N$, define global probe mass:

$$M_\Omega(z) = \sum_{i\in\Omega} softmax(z)_i.$$

In the empirical implementation, we additionally use coordinate-wise probe mass:

$$M_\Omega^{coord}(z) = \sum_{i\in\Omega} \frac{\exp z_i}{\sum_{j\in G(\Omega)} \exp z_j},$$

where $G(\Omega)$ is the union of competing probes within the same coordinate. This prevents outcome, role, and stance probes from competing against each other globally.

The interpreted layer state is:

$$\begin{aligned} m_t^{(\ell)}(c,u) = \big(&M_{\Omega_O(u)}(z_t^{(\ell)}(c,u)), \\ &M_{\Omega_R(c,u)}(z_t^{(\ell)}(c,u)), \\ &M_{\Omega_A(c,u)}(z_t^{(\ell)}(c,u))\big). \end{aligned}$$

The interpreted trajectory is:

$$\begin{aligned} \widetilde{T}_t(c,u) = \big(&m_t^{(0)}(c,u), m_t^{(1)}(c,u), \\ &\ldots, m_t^{(L)}(c,u)\big). \end{aligned}$$

L Outcome Persistence and Role/Stance Divergence Proofs

Definition D.1: Interpretive success. For input (c, u), the model is interpretively successful at target token t if, at the final layer, it assigns nonzero mass to an outcome-compatible probe set and separates the relevant role/stance probe sets from incompatible alternatives:

$$M_{\Omega_O(u)}(z_t^{(L)}(c, u)) > 0.$$

Theorem D.1: Outcome Support Persistence. Let (c_1, u) and (c_2, u) be outcome-stable:

$$O(c_1, u) = O(c_2, u) = O(u).$$

If the model is interpretively successful on both inputs, then:

$$M_{\Omega_O(u)}(z_t^{(L)}(c_1, u)) > 0,$$
$$M_{\Omega_O(u)}(z_t^{(L)}(c_2, u)) > 0.$$

Proof. Since the pair is outcome-stable, both contexts share the same outcome-compatible probe set $\Omega_O(u)$. By interpretive success, each final predictive state assigns nonzero mass to the outcome-compatible set. Therefore both inequalities hold.

Theorem D.2: Role/Stance-Sensitive Final Divergence. Let (c_1, u) and (c_2, u) be outcome-stable:

$$O(c_1, u) = O(c_2, u) = O(u),$$

but suppose:

$$R(c_1, u) \neq R(c_2, u) \quad \text{or} \quad A(c_1, u) \neq A(c_2, u).$$

If the model is interpretively successful on both inputs, then:

$$m_t^{(L)}(c_1, u) \neq m_t^{(L)}(c_2, u).$$

Proof. Assume for contradiction:

$$m_t^{(L)}(c_1, u) = m_t^{(L)}(c_2, u).$$

Then all ORA coordinate masses are equal. But if $R(c_1, u) \neq R(c_2, u)$, interpretive success requires separating the role-sensitive alternatives. If $A(c_1, u) \neq A(c_2, u)$, interpretive success requires separating the stance-sensitive alternatives. Identical interpreted final states cannot separate these incompatible alternatives. Contradiction. Hence:

$$m_t^{(L)}(c_1, u) \neq m_t^{(L)}(c_2, u).$$

Theorem D.3: Context-Sensitive Logit Trajectory Necessity. If an ORA pair has fixed utterance form, stable outcome space, and different role or stance assignments, then successful interpretation requires:

$$T_t(c_1, u) \neq T_t(c_2, u)$$

in an ORA probe-relevant direction. Equivalently, there exists some layer ℓ and some probe set:

$$\Omega \in \{\Omega_R, \Omega_A\}$$

such that:

$$M_\Omega(z_t^{(\ell)}(c_1, u)) \neq M_\Omega(z_t^{(\ell)}(c_2, u)).$$

Proof. By Theorem D.2:

$$m_t^{(L)}(c_1, u) \neq m_t^{(L)}(c_2, u).$$

Suppose, for contradiction, that the interpreted trajectories are identical:

$$\widetilde{T}_t(c_1, u) = \widetilde{T}_t(c_2, u).$$

Then every layerwise interpreted state is identical, including the final state:

$$m_t^{(L)}(c_1, u) = m_t^{(L)}(c_2, u),$$

contradicting Theorem D.2. Therefore:

$$\widetilde{T}_t(c_1, u) \neq \widetilde{T}_t(c_2, u).$$

Each interpreted state is a function of layerwise logits. Thus, if interpreted trajectories differ, there is at least one layer ℓ at which some probe mass differs. Since the outcome is stable, the relevant divergence must lie in role or stance coordinates. Therefore the logit trajectories differ in an ORA-relevant direction.

M Context Sensitivity and Jacobian Necessity

Let:

$$x_1 = (c_1, u), \qquad x_2 = (c_2, u)$$

share the same target utterance u. Suppose:

$$\widetilde{T}_t(c_1, u) \neq \widetilde{T}_t(c_2, u).$$

Assume each layer map is differentiable in a neighborhood of the relevant hidden states. Define the context-token Jacobian block:

$$J_{t,j}^{(\ell)} = \frac{\partial z_t^{(\ell+1)}}{\partial h_j^{(\ell)}}, \qquad j \leq m.$$

Theorem D.4: Jacobian Necessity of Context Sensitivity. If all context-token Jacobian blocks are zero:

$$J_{t,j}^{(\ell)} = 0$$

for every layer ℓ and every context position $j \leq m$, then context-token hidden states cannot affect the target-token output-space trajectory. Therefore, successful ORA-sensitive interpretation requires some layer ℓ and context position $j \leq m$ such that:

$$J_{t,j}^{(\ell)} \neq 0.$$

Proof sketch. If all context-token Jacobian blocks are zero, then infinitesimal perturbations to context-token hidden states have no effect on target-token logits at the next layer. By induction over layers, the target-token output-space trajectory is invariant to context-token hidden states. But ORA-sensitive interpretation requires:

$$\widetilde{T}_t(c_1, u) \neq \widetilde{T}_t(c_2, u).$$

Therefore some context-token hidden state must affect the target-token output-space trajectory. Hence some $J_{t,j}^{(\ell)} \neq 0$.

This theorem establishes a minimal dependence condition. It does not claim that attention weights are explanations, nor that attention is the only possible carrier of context. It only states that context must influence the computation.

N Bounded State Interpreter Formalization

Theorem E.1: Transformer State-Transition Trace. For any input $x = (c, u)$, a transformer with finite depth L defines a finite state-transition computation:

$$H^{(0)}(x) \to H^{(1)}(x) \to \cdots \to H^{(L)}(x),$$

where:

$$H^{(\ell+1)}(x) = F_\theta^{(\ell)}(H^{(\ell)}(x)).$$

The target-token logit trajectory:

$$T_t(x) = \big(z_t^{(0)}(x), z_t^{(1)}(x), \ldots, z_t^{(L)}(x)\big)$$

is the output-space readout of that hidden-state transition trace.

Proof. The transformer receives the embedded sequence:

$$H^{(0)}(x) = (h_1^{(0)}, \ldots, h_T^{(0)}).$$

For each layer ℓ, it applies a deterministic map $F_\theta^{(\ell)}$:

$$H^{(\ell+1)}(x) = F_\theta^{(\ell)}(H^{(\ell)}(x)).$$

Since L is finite, this yields a finite sequence:

$$H^{(0)}(x), H^{(1)}(x), \ldots, H^{(L)}(x).$$

For a fixed target token t, each layer has a hidden vector:

$$h_t^{(\ell)}(x) \in \mathbb{R}^d.$$

Applying the fixed output projection:

$$z_t^{(\ell)}(x) = h_t^{(\ell)}(x) W_U.$$

Therefore $T_t(x)$ is obtained by applying the same readout to the target-token hidden state at each layer. By Theorem C.1, every hidden-state displacement induces an exact output-space displacement. Thus the logit trajectory is the output-space projection of the hidden-state transition trace.

Definition E.1: Bounded State Interpreter. A bounded state interpreter is a tuple:

$$B = (S, I, O, U, R, T_B),$$

where S is a state space, I is an input/context space, O is an output/readout space, U is a transition map, R is a readout map, and $T_B < \infty$ is a finite transition horizon.

For input $i \in I$, the system initializes:

$$s_0 = \Phi(i),$$

evolves a finite trajectory:

$$s_0 \to s_1 \to \cdots \to s_{T_B}$$

under:

$$s_{k+1} = U_k(s_k),$$

and produces readouts:

$$o_k = R(s_k).$$

It is an interpreter when the input determines not merely the content being processed, but the trajectory by which the system resolves that content. It is bounded because state dimension, input context, and transition horizon are finite.

Theorem E.2: Bounded State Interpreter Theorem for ORA-Resolving Transformers. Let T_θ be a transformer with finite depth L, finite context length T, hidden dimension d, layer maps $F_\theta^{(\ell)}$, and output projection W_U. Suppose T_θ successfully resolves an ORA pair $(c_1, u), (c_2, u)$ such that:

$$O(c_1, u) = O(c_2, u) = O(u),$$

but:

$$R(c_1, u) \neq R(c_2, u) \quad \text{or} \quad A(c_1, u) \neq A(c_2, u).$$

Then T_θ functions as a bounded state interpreter over ORA contexts.

Proof. For fixed input length T and hidden dimension d:

$$H^{(\ell)}(x) \in \mathbb{R}^{T \times d}.$$

Let:

$$S = \mathbb{R}^{T \times d}.$$

The input $x = (c, u)$ initializes:

$$H^{(0)}(x) = (E[x_1] + p_1, \ldots, E[x_T] + p_T) = \Phi_\theta(c, u).$$

The transformer has finitely many layers:

$$H^{(\ell+1)}(x) = F_\theta^{(\ell)}(H^{(\ell)}(x)),$$

so it evolves:

$$H^{(0)}(x) \to H^{(1)}(x) \to \cdots \to H^{(L)}(x).$$

For target token t, define:

$$R_t(H^{(\ell)}(x)) = h_t^{(\ell)}(x) W_U = z_t^{(\ell)}(x).$$

The ORA interpreted state is:

$$m_t^{(\ell)}(c, u) = \big(M_{\Omega_O(u)}(z_t^{(\ell)}), M_{\Omega_R(c,u)}(z_t^{(\ell)}), M_{\Omega_A(c,u)}(z_t^{(\ell)})\big).$$

By Theorem D.3:

$$\widetilde{T}_t(c_1, u) \neq \widetilde{T}_t(c_2, u).$$

Therefore the two contexts induce distinct interpreted trajectories. Since interpreted trajectories are functions of logit trajectories, and logit trajectories are readouts of hidden-state trajectories, the contexts induce distinct ORA-relevant state trajectories. The transformer therefore has finite-dimensional states, input-conditioned initialization, finite transition horizon, logit readout, ORA interpreted readout, and context-sensitive trajectory divergence. Hence it satisfies the bounded state interpreter definition.

O Turing-Machine-Like but not Classical Turing Machine

Corollary E.1. An ORA-resolving transformer is not a classical Turing machine, but it is Turing-machine-like at the level of bounded configuration dynamics.

Proof. A classical Turing machine evolves configurations:

$$C_0 \to C_1 \to \cdots$$

where:

$$C_t = (q_t, \tau_t, i_t)$$

and:

$$C_{t+1} = \delta(C_t).$$

By Theorem E.2, an ORA-resolving transformer evolves finite hidden-state configurations:

$$H^{(0)} \to H^{(1)} \to \cdots \to H^{(L)}$$

under layer maps:

$$H^{(\ell+1)} = F_\theta^{(\ell)}(H^{(\ell)}).$$

It reads out logits:

$$z_t^{(\ell)} = h_t^{(\ell)} W_U$$

and interpreted probe masses:

$$m_t^{(\ell)} = (M_{\Omega_O}, M_{\Omega_R}, M_{\Omega_A}).$$

Thus the analogy holds at the level of state, transition, readout, and finite trace. However, the transformer lacks unbounded tape, unbounded computation time, exact symbolic configurations, and a discrete transition table. Its trajectory is finite, continuous, high-dimensional, and bounded by context length and model depth. Therefore it is not a classical Turing machine. It is a bounded state interpreter.

P Probe Set Construction

Each ORA example is represented as:

`task`, `surface`, `left_context`, `right_context`, `O`, `R_left`, `R_right`, `A_left`, `A_right`.

The left and right contexts instantiate the same utterance form under different discourse conditions. O contains outcome-compatible probes. R_{left} and R_{right} contain competing role probes. A_{left} and A_{right} contain competing stance probes. Table 6 gives a representative probe-set example.

Probe words are mapped to first-token ids using the model tokenizer. This is a lightweight logit-lens approximation. For coordinate-wise softmax, the competing role group is:

$$G_R = R_{\text{left}} \cup R_{\text{right}},$$

and the competing stance group is:

$$G_A = A_{\text{left}} \cup A_{\text{right}}.$$

Mass is computed only within the relevant coordinate group.

Q Trajectory Extraction Details

All ORA logit-lens experiments use `distilgpt2`. For each prompt, we extract all hidden states with `output_hidden_states=True`. The prompt template is:

```
Context: {context}
Utterance: "{surface}"
The pragmatic interpretation of the utterance is
```

The target position is the final prompt token. For each layer, we extract $h_t^{(\ell)}$. When the model exposes a final layer norm, we apply it before the LM head:

$$z_t^{(\ell)} = h_t^{(\ell)} W_U.$$

The result bundle is organized into five directories: `outputs/`, `outputs_sweep/`, `outputs_geometry/`, `outputs_automata/`, and `large_json_experiment/`. Full file names are listed in Appendix Y.

R Full ORA Logit-Lens Results

The full hand-specified ORA set evaluates whether same-form utterance pairs preserve outcome mass while diverging in role and stance coordinates. Table 12 reports the full summary metrics. Table 7 reports a context/self attention diagnostic used only as a sanity check. Figure 1a shows the complete layerwise ORA probe trajectories.

S Candidate Sweep Results

The candidate sweep tests whether the same ORA geometry appears beyond the hand-specified items. Each candidate preserves surface form across contexts, uses stable outcome probes, and varies role or stance probes. Table 13 gives the full ranking. Figure 1b shows the top trajectory cases.

T Trajectory Geometry Metrics

Let $x_{\text{left}}^{(\ell)}$ and $x_{\text{right}}^{(\ell)}$ be the layerwise ORA coordinate vectors for the two contexts. Role divergence is:

$$D_R^{(\ell)} = \left\| x_{R,\text{left}}^{(\ell)} - x_{R,\text{right}}^{(\ell)} \right\|_2.$$

Stance divergence is:

$$D_A^{(\ell)} = \left\| x_{A,\text{left}}^{(\ell)} - x_{A,\text{right}}^{(\ell)} \right\|_2.$$

Combined role/stance divergence is:

$$D_{RA}^{(\ell)} = \left\| x_{RA,\text{left}}^{(\ell)} - x_{RA,\text{right}}^{(\ell)} \right\|_2.$$

Bifurcation depth is:

$$\ell^* = \min\{\ell : D_{RA}^{(\ell)} > \tau\}, \qquad \tau = 0.35.$$

Trajectory curvature is:

$$\kappa^{(\ell)} = \left\| x^{(\ell+1)} - 2x^{(\ell)} + x^{(\ell-1)} \right\|_2.$$

Table 14 reports the full geometry metrics. Figures 3–**??** show the summary, divergence curves, and curvature curves.

U Bounded Automata-State Probe

We use a controlled parenthesis-depth task to test bounded latent-state tracking. We generate 700 random sequences over parentheses, with lengths between 12 and 32, and maximum depth 6. The true latent state is:

$$d_t = \#\text{ unmatched left parentheses after token } t.$$

Depth is clipped to $d_t \in [0, 6]$. For each parenthesis token, we extract hidden states from every layer. For each layer, we train a linear probe:

$$\hat{d}_t = \arg\max W h_t^{(\ell)} + b$$

to predict d_t. We use an $80/20$ train/test split and train for 80 epochs.

Transition consistency tests whether decoded states obey:

$$d_{t+1} = \begin{cases} \min(d_t + 1, 6), & x_{t+1} = \text{“(”} \\ \max(d_t - 1, 0), & x_{t+1} = \text{“)”}. \end{cases}$$

Table 8 reports depth decoding and transition consistency. Figures **??**–5 show the corresponding probe and trajectory plots.

V Prompt–Response Regime Operator Analysis

The prompt–response operator analysis tests whether bounded trajectory structure appears beyond hand-built ORA examples. The pipeline is:

- `00_prepare_pairs.py`: extract pairs and labels.
- `01_extract_features.py`: extract prompt/response states.
- `02_probe_and_operator.py`: train probes and operators.
- `03_regime_operators.py`: fit regime operators.
- `04_regime_operators_pca.py`: fit rank-32 PCA operators.

For each layer, $h_{\text{prompt}}^{(\ell)}$ and $h_{\text{response}}^{(\ell)}$ are extracted. We fit:

$$h_{\text{response}}^{(\ell)} \approx A_r^{(\ell)} h_{\text{prompt}}^{(\ell)}$$

using ridge regression. The relative error is:

$$\epsilon = \frac{\|H_{\text{response}} - \widehat{H}_{\text{response}}\|_F}{\|H_{\text{response}}\|_F}.$$

Table 9 reports label counts. Tables 15–11 report operator and probe summaries. Figures **??**–**??** show probe and operator curves.

W Full Appendix Tables

Utterance	Context	Outcome $O(u)$	Role $R(c, u)$	Stance $A(c, u)$
chāvi kāḍ	Door is locked	Access achieved by key / lock opened	Addressee as agent using the key	Instruction
chāvi kāḍ	Keys are needed from addressee	Key becomes available	Addressee as holder/source or affected participant	Request
roṭṭi ta	Kitchen context	Rotis made and brought	Addressee as cook/fetcher/direct agent	Instruction
roṭṭi ta	Store/procurement context	Rotis brought/procured	Addressee as buyer/procurer	Delegation
roṭṭi ta	Event/delegation context	Rotis appear / are arranged	Addressee as coordinator	Command/directive
file khulvāo	Clerk context	File opened or processed	Addressee as direct executor	Command
file khulvāo	Manager/authority context	File opened or processed	Addressee as coordinator/authority	Directive / authorization request
khulvāne-kā sīkh	Instructional context	Opening action performed or learned	Addressee as agent / incompetent agent	Humiliative teaching
khulvāne-kā sīkh	Abusive context	Abstract opening/causation frame	Addressee as target of evaluation	Dominance / insult
fuck you	Close friends joking	Direct affective address	Addressee as intimate target / co-participant in banter	Playful aggression / solidarity
fuck you	Hostile argument	Direct affective address	Addressee as target of rejection	Hostility / rejection / attack

Table 5: Full ORA inventory for representative utterances. The utterance-local form constrains the outcome or force domain, while role and stance are resolved in discourse.

Field	Value
surface	*fuck you*
O	intense, emotional, direct
R_{left}	playful, affectionate, teasing
R_{right}	hostile, angry, rejecting
A_{left}	tease, joke, flirt
A_{right}	reject, attack, end

Table 6: Example ORA probe construction for *fuck you*.

Layer	Attention to Context	Attention to Target Self
1	0.220	0.780
2	0.123	0.877
3	0.029	0.971
4	0.000	1.000

Table 7: Mean context/self attention diagnostic from `outputs/ora_attention.csv`. This is a context-flow sanity check, not an attention-as-explanation claim.

Layer	Test Acc.	Test MAE	Transition Acc.	Transition MAE
0	0.466	1.378	0.530	1.187
1	0.931	0.069	0.900	0.100
2	0.951	0.049	0.903	0.097
3	0.998	0.002	0.994	0.006
4	0.975	0.025	0.963	0.037
5	0.970	0.030	0.954	0.048
6	0.953	0.047	0.938	0.062

Table 8: Depth decoding and transition consistency for the bounded parenthesis automaton task.

Label Type	Label	Count
prompt task	general	2615
prompt task	code	2213
prompt task	writing	145
prompt task	emotional	26
prompt task	theory	1
response family	code	3375
response family	answer	1341
response family	clarification	225
response family	stepwise	50
response family	emotional support	7
response family	refusal	2

Table 9: Label counts for the 5000-example prompt–response feature subset.

Task	Best Layer	Best Error	Mean Error
code	1	0.369	0.413
general	6	0.404	0.997
writing	0	0.325	71.234

Table 10: Full-space prompt-task operator summary. The unstable writing mean reflects poor conditioning in the unreduced full-space fit; the rank-32 PCA version is the preferred operator analysis.

Label	Position	Best Layer	Train Acc.	Test Acc.
prompt task	prompt end	5	0.900	0.868
prompt task	response end	6	0.786	0.733
response family	prompt end	1	0.704	0.704
response family	response end	6	0.840	0.819

Table 11: Best linear probe accuracies for prompt-task and response-family labels.

Task	D_O	D_{RA}	O_L	O_R	R_L/R_R	A_L/A_R
chavi_kadh	0.000	2.951	1.000	1.000	0.988	0.988
rotti_ta	0.000	2.951	1.000	1.000	0.988	0.988
file_khulvao	0.000	2.951	1.000	1.000	0.988	0.988
khulvane_ka_sikh	0.000	2.951	1.000	1.000	0.988	0.988
fuck_you	0.000	2.951	1.000	1.000	0.988	0.988
goddamnit	0.000	2.951	1.000	1.000	0.988	0.988
damn_you	0.000	2.951	1.000	1.000	0.988	0.988
motherfucking_bastard	0.000	2.951	1.000	1.000	0.988	0.988

Table 12: Full ORA logit-lens summary from `outputs/ora_summary_metrics.csv`. D_O is outcome distance and D_{RA} is role/stance trajectory distance.

Task	D_O	D_R	D_A	D_{RA}	ORA Score
fuck_you	0.000	0.607	0.254	0.658	0.658
clear_the_case	0.000	0.640	0.072	0.644	0.644
go_to_hell	0.000	0.519	0.197	0.555	0.555
you_monster	0.000	0.421	0.270	0.500	0.500
you_idiot	0.000	0.434	0.066	0.439	0.439
damn_you	0.000	0.402	0.081	0.410	0.410
i_hate_you	0.000	0.347	0.006	0.347	0.347
release_the_payment	0.000	0.321	0.036	0.323	0.323
get_out	0.000	0.186	0.240	0.304	0.304
khulvane_ka_sikh	0.000	0.255	0.064	0.263	0.263
shut_up	0.000	0.235	0.097	0.254	0.254
goddamnit	0.000	0.175	0.065	0.186	0.186
send_it	0.000	0.034	0.117	0.122	0.122
open_the_file	0.000	0.106	0.001	0.106	0.106
chavi_kadh	0.000	0.095	0.021	0.097	0.097
file_khulvao	0.000	0.083	0.005	0.084	0.084
make_it_happen	0.000	0.062	0.015	0.064	0.064

Table 13: Full ORA candidate-sweep ranking from `outputs_sweep/ora_candidate_rankings.csv`. D_O is outcome drift; D_R, D_A, D_{RA} are role, stance, and combined role/stance distances. ORA score is $D_{RA} - D_O$.

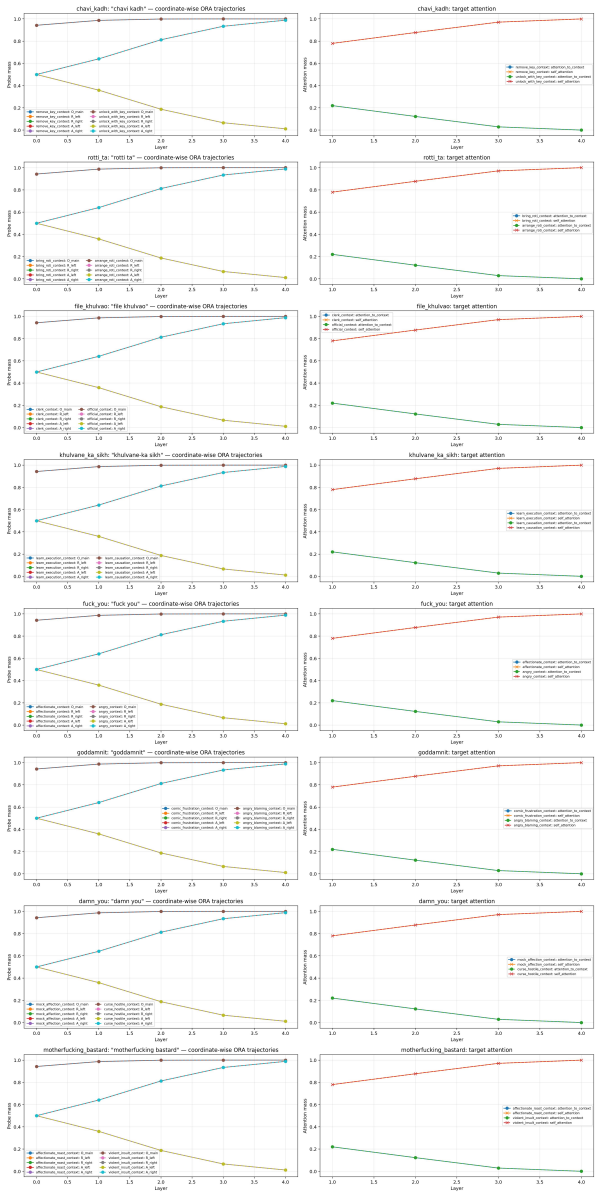
Task	Class	Bif.	Final D_{RA}	AUC(D_R)	AUC(D_A)	AUC(D_{RA})
fuck_you	affective	5	0.113	1.297	0.397	1.405
clear_the_case	institutional	2	0.047	1.352	0.098	1.360
you_monster	affective	5	0.078	0.861	0.544	1.069
go_to_hell	affective	5	0.120	0.871	0.386	1.022
damn_you	affective	–	0.069	0.759	0.044	0.787
you_idiot	affective	5	0.096	0.745	0.091	0.783
get_out	affective	–	0.076	0.302	0.407	0.626
release_the_payment	institutional	–	0.044	0.544	0.049	0.558
i_hate_you	affective	–	0.048	0.498	0.009	0.498
shut_up	affective	–	0.074	0.411	0.160	0.462
khulvane_ka_sikh	indic/mixed	–	0.008	0.291	0.037	0.298
goddamnit	affective	–	0.063	0.244	0.072	0.275
send_it	institutional	–	0.014	0.068	0.188	0.220
open_the_file	institutional	–	0.045	0.210	0.001	0.210
chavi_kadh	indic/mixed	–	0.023	0.197	0.015	0.203
make_it_happen	institutional	–	0.013	0.126	0.016	0.131
file_khulvao	indic/mixed	–	0.020	0.102	0.003	0.103

Table 14: Full ORA trajectory geometry metrics from `outputs_geometry/ora_geometry_metrics.csv`.

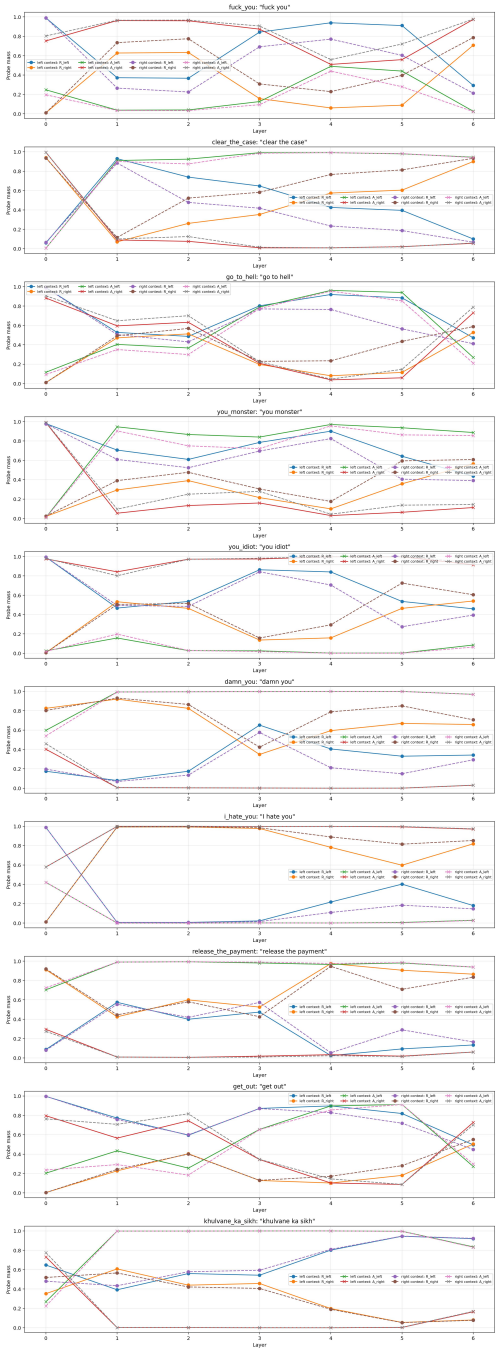
Regime Type	Regime	Best Layer	Best Error	Final Error	Mean Error	PCA Expl.	N
response family	clarification	6	0.562	0.562	0.629	0.874	225
prompt task	code	5	0.573	0.577	0.633	0.670	2213
prompt task	writing	4	0.656	0.695	0.693	0.823	145
response family	answer	5	0.699	0.760	0.752	0.781	1341
global	GLOBAL	5	0.736	0.752	0.775	0.733	5000
response family	code	5	0.742	0.746	0.776	0.717	3375
prompt task	general	3	0.833	0.913	0.859	0.793	2615

Table 15: Rank-32 PCA prompt–response operator summary. Several regime-specific operators outperform the global operator.

X Full Appendix Figures



(a) Full ORA combined logit-lens trajectories for the hand-specified ORA items. Each panel shows layerwise probe-mass movement for the same surface form under two contexts.



(b) Top ORA candidate trajectories from the sweep. The plot shows role and stance probe-mass trajectories for the highest-scoring same-form/different-context pairs.

Figure 1: Full appendix ORA trajectory figures. Left: hand-specified ORA items. Right: top ORA candidates from the sweep.

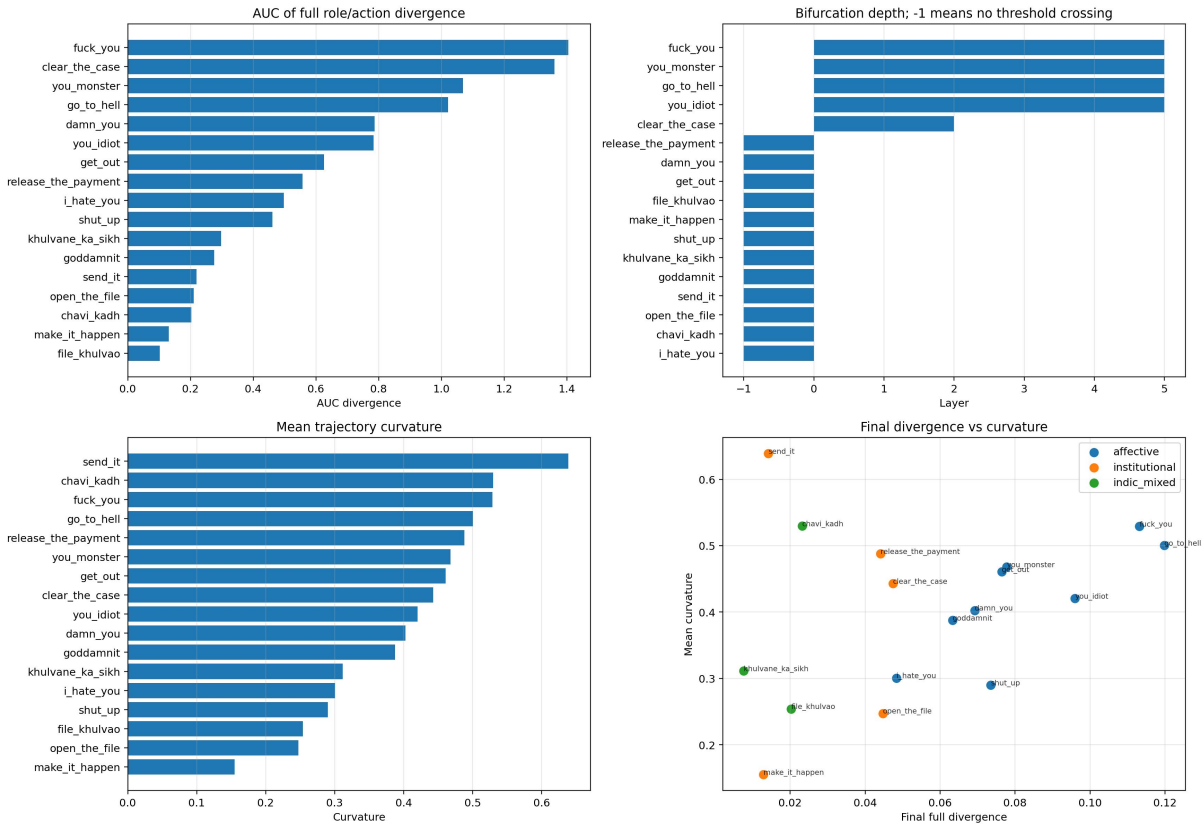
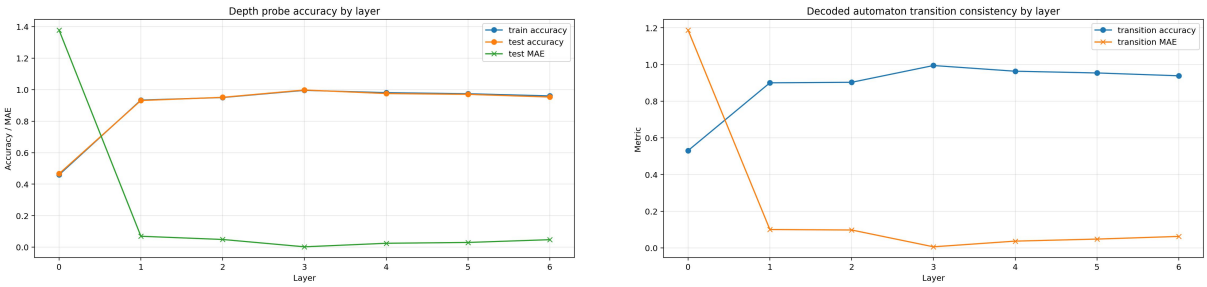


Figure 3: ORA geometry summary: AUC of role/stance divergence, bifurcation depth, curvature, and final divergence/curvature scatter.



(a) Depth probe accuracy and MAE.

(b) Transition consistency.

Figure 4: Bounded automata-state probe diagnostics.

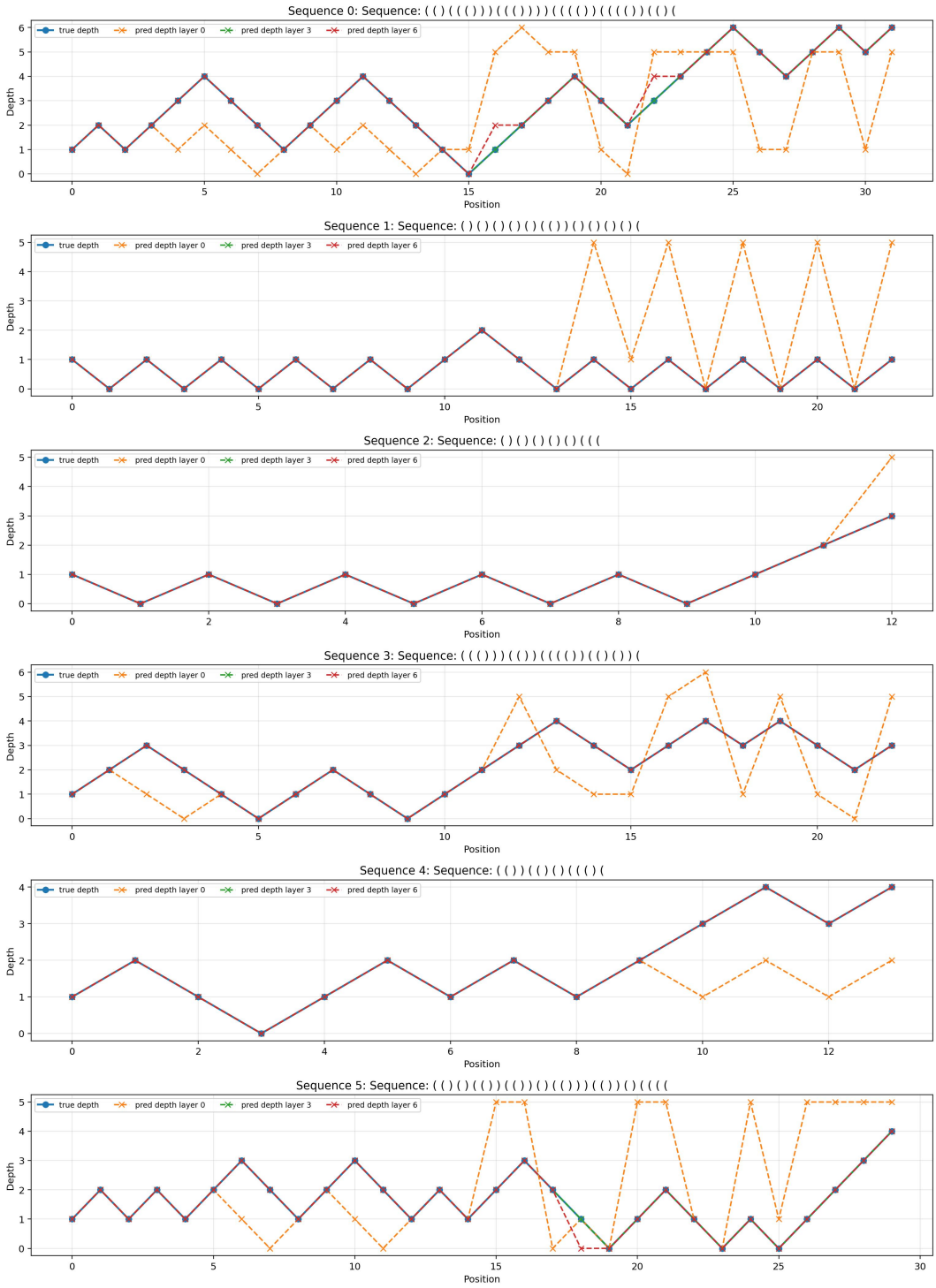
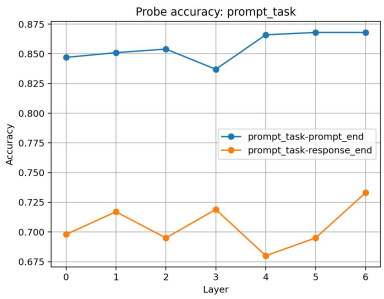
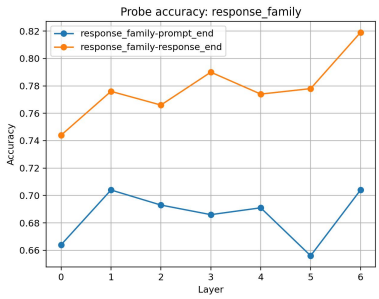


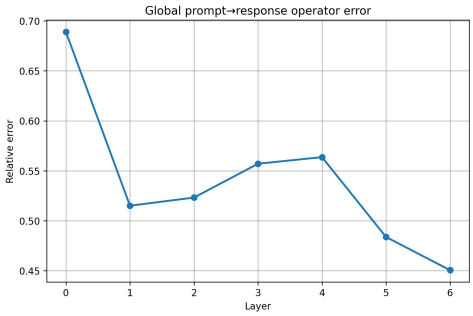
Figure 5: Representative true and decoded bounded-depth state trajectories.



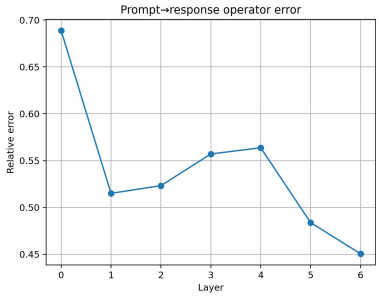
(a) Prompt-task probe accuracy.



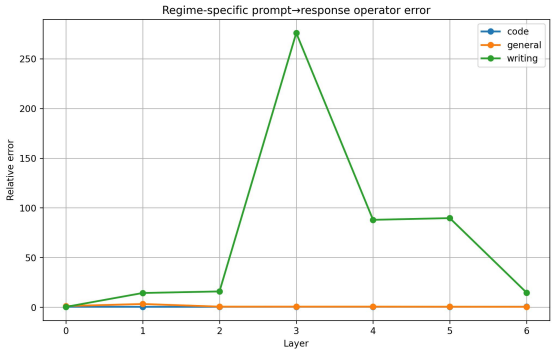
(b) Response-family probe accuracy.



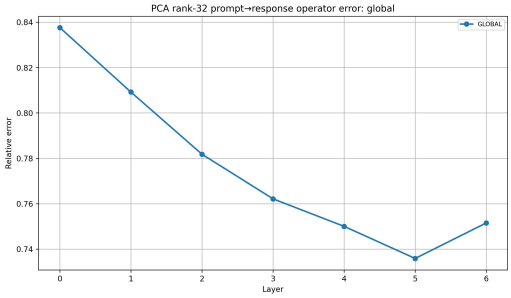
(c) Global operator error.



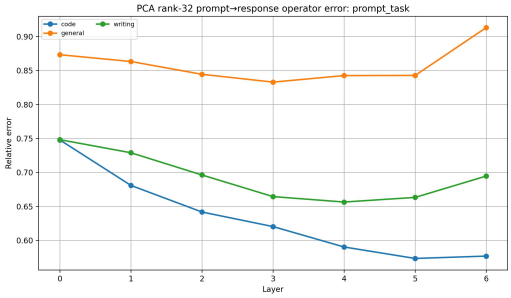
(d) Transition operator error.



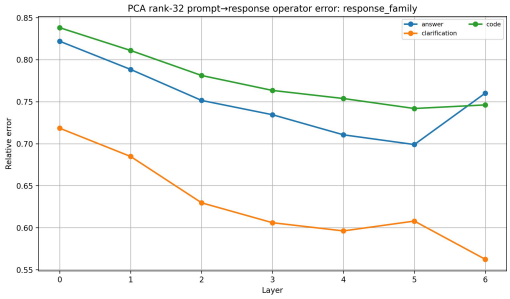
(e) Regime-specific prompt-to-response operator error curves.



(f) Global PCA operator error.



(g) Prompt-task PCA operator error.



(h) Response-family PCA operator error.

Figure 6: Prompt–response probe and operator diagnostics. Top: probe accuracy and operator-error curves. Middle: regime-specific prompt-to-response operator error. Bottom: rank-32 PCA prompt-to-response operator errors.

Y Result Files Included in the Artifact Bundle

The complete result bundle includes the following files:

- `outputs/ora_attention.csv`
- `outputs/ora_probe_trajectories.csv`
- `outputs/ora_raw_logits.csv`
- `outputs/ora_summary_metrics.csv`
- `outputs/ora_combined_plots.jpg`
- `outputs_sweep/ora_candidate_rankings.csv`
- `outputs_sweep/ora_probe_trajectories.csv`
- `outputs_sweep/ora_raw_probe_logits.csv`
- `outputs_sweep/top_ora_candidates.jpg`
- `outputs_geometry/ora_geometry_metrics.csv`
- `outputs_geometry/ora_layerwise_divergence.csv`
- `outputs_geometry/ora_geometry_summary.jpg`
- `outputs_geometry/ora_divergence_curves.jpg`
- `outputs_geometry/ora_curvature_curves.jpg`
- `outputs_automata/generated_parentheses_dataset.csv`
- `outputs_automata/true_depth_trajectories.csv`
- `outputs_automata/depth_probe_accuracy.csv`
- `outputs_automata/depth_transition_metrics.csv`
- `outputs_automata/depth_state_trajectories.csv`
- `large_json_experiment/analysis/probe_results.csv`
- `large_json_experiment/analysis/regime_operator_errors.csv`
- `large_json_experiment/analysis/regime_operator_summary.csv`
- `large_json_experiment/analysis/regime_operator_pca_errors.csv`
- `large_json_experiment/analysis/regime_operator_pca_summary.csv`
- `large_json_experiment/features/logit_probe_masses.csv`
- `large_json_experiment/features/feature_metadata.csv`

Z Ablations and Controls

The current experiments include three diagnostic control families.

Outcome drift control. Candidate ranking subtracts outcome distance:

$$score_{ORA} = D_{RA} - D_O.$$

This prevents items from ranking highly merely because the outcome coordinate changes.

Coordinate-local softmax. ORA probe masses are computed within coordinate groups rather than over the full vocabulary. This controls for unrelated high-probability vocabulary items and isolates movement within role and stance alternatives.

Regime/operator control. The prompt–response operator experiments include a global operator and regime-specific operators. The fact that several regime-specific operators outperform the global operator suggests that transition structure is not arbitrary hidden-state drift.

Future controls should include native-script multilingual probes, random probe sets, context removal, neutral contexts, phrase-level probes, and same-context sanity checks.